|  |  |
| --- | --- |
| Amelio-Cypress Automation Framework | Amélio |

|  |  |  |  |
| --- | --- | --- | --- |
| # | Module | Features | Done? |
| 1 | Test Suites | UI, End-to-End, Smoke, Sanity, Regression, API, Database | HD Green Check True Tick Mark Icon Sign PNG |
|  |  |  |  |
| 2 | Reports | Customized Report (Local), Cypress Cloud Reports (Advanced) | HD Green Check True Tick Mark Icon Sign PNG |
|  |  |  |  |
| 3 | Error Screenshots | Every Failed testcase get Error Screenshot captured | HD Green Check True Tick Mark Icon Sign PNG |
|  |  |  |  |
| 4 | Log File | Implemented loggers in all tests | HD Green Check True Tick Mark Icon Sign PNG |
|  |  |  |  |
| 5 | Repository | Git/GitHub project Repository added | HD Green Check True Tick Mark Icon Sign PNG |
|  |  |  |  |
| 6 | CI/CD | AzureDevops CI integrated for Cypress tests run on each new commit | HD Green Check True Tick Mark Icon Sign PNG |
|  |  |  |  |
| 7 | Docker | For creating different environments, test setup, different Users with permissions | HD Green Check True Tick Mark Icon Sign PNG |
|  |  |  |  |
| 8 | Parallelization | Running tests parallel in multiple machines using Cypress cloud | HD Green Check True Tick Mark Icon Sign PNG |
|  |  |  |  |
| 9 | Data Driven | Test data reading from Excel files | HD Green Check True Tick Mark Icon Sign PNG |
|  |  |  |  |
| 10 | BDD-Cucumber | Implemented Cucumber for not technical stakeholders navigate through testcases (add, modify) | HD Green Check True Tick Mark Icon Sign PNG |
|  |  |  |  |
| 11 | Environments | Implemented CI Pipelines to run tests in Test, Stage, Production environments as per schedules |  |
|  |  |  |  |
| 12 | Documentation | Creating onboarding documents, videos for quick start with using automation framework |  |
|  |  |  |  |
| 13 | Cross Browser | Integrated with all the browsers currently supported by Cypress |  |
|  |  |  |  |
| 14 | Test Case Management | Integration with test case management tools (e.g., TestRail, Zephyr) for better traceability. Results added on each run |  |
|  |  |  |  |
| 15 | Mocking and Stubbing | Use of Cypress features to mock API responses for testing without relying on external services |  |
|  |  |  |  |
| 16 | Test/ code coverage tools | Using Istanbul for getting percentage of Testcases covered (applies to both Unit tests, Automation tests) |  |
|  |  |  |  |
| 17 | visual testing tools | BackstopJS – Cypress integration for visual regression testing |  |
|  |  |  |  |
| 18 | Version Control for Test Data | Version control for test data to ensure consistent test runs | HD Green Check True Tick Mark Icon Sign PNG |
|  |  |  |  |
| 19 | Browser and Device Emulation | Using Cypress Native Device Emulation (cypress-device-emulator)  Options to test responsiveness on different screen sizes and devices. |  |
| 20 | Test Filtering and Tagging | Using plug-in cypress-grep (open source) |  |
|  |  |  |  |
| 21 | Run on Real Devices | Integration with BrowserStack to run tests on real devices |  |
|  |  |  |  |
| 22 | Test Scheduling | Azure DevOps pipeline – corn Jobs |  |
|  |  |  |  |
| 23 | Performance Monitoring Tools | Using Datadog, Dynatrace for performance monitoring |  |
|  |  |  |  |
| 24 | Log Monitoring and Analysis Tools | Elastic Stack (ELK) to analyze logs generated during Cypress tests. You can send logs from your tests to the ELK stack for visualization and analysis. |  |
|  |  |  |  |
|  |  |  |  |
|  |  | **Cypress Cloud Features Integrated** |  |
|  |  |  |  |
| 1 | Amelio Features coverage | Test Suite Size - Tag (feature name - e.g. Survey) Test Coverage percentage | HD Green Check True Tick Mark Icon Sign PNG |
|  |  |  |  |
| 2 | Flaky tests Identification | Flaky tests detection (Test Failed and re-run then passed) | HD Green Check True Tick Mark Icon Sign PNG |
|  |  |  |  |
| 3 | Test Errors | Most Common Errors, Slowest Tests | HD Green Check True Tick Mark Icon Sign PNG |
|  |  |  |  |
| 4 | UI Coverage | Page by Page shows test percentage covered |  |
|  |  |  |  |
|  |  |  |  |
|  |  | **Testing Types Covered** |  |
|  |  |  |  |
| 1 | Functional | End to End test cases covering the functionalities. |  |
| 2 | UI | Covers look and feel of the application |  |
| 3 | API | Extensively covered all API with combination of input requests |  |
| 4 | Database | Testing responses stored in DB, includes schema validations |  |
| 5 | Unit Testing | Using Jest, mocha, Chai tools to build robust unit test cases |  |
| 6 | Integration Testing | Testing the interaction between integrated components or systems. |  |
| 7 | Smoke Testing | Basic tests to check if the major functions work without going into detail |  |
| 8 | Sanity Testing | A subset of regression testing focused on verifying specific functionalities after changes |  |
| 9 | Regression Testing | Testing to ensure that new code changes do not adversely affect existing functionality |  |
| 10 | Performance Testing | Evaluating the speed, scalability, and stability of the application under various conditions.  **1.Load Testing**: Testing the system's behaviour under expected load.  **2.Stress Testing**: Testing beyond normal operational capacity to determine breaking points.  3.**Endurance Testing**: Testing how the system performs under a sustained load over time.  4.**Spike Testing**: Testing the system's reaction to sudden large spikes in load |  |
| 11 | Security Testing | Identifying vulnerabilities, threats, and risks in the software and ensuring that data and resources are protected.  **OWASP ZAP (Zed Attack Proxy)**: A widely-used open-source DAST tool for finding vulnerabilities in web applications. |  |
| 12 | Compatibility Testing | Testing the application across different browsers, devices, operating systems, and network environments. |  |
| 13 | Localization and Internationalization Testing | Ensuring the application can adapt to various languages and regional differences. |  |
| 14 | Mobile Testing | Testing applications on mobile devices for performance, usability, and compatibility |  |